Gearbox modeling and simulation

Can transient loads occur internally in a planetary gearbox? How to simulate external modules fully coupled with HAWC2

Larsen, Torben J.; Rasmussen, Flemming; Hansen, Anders Melchior

Publication date:
2012

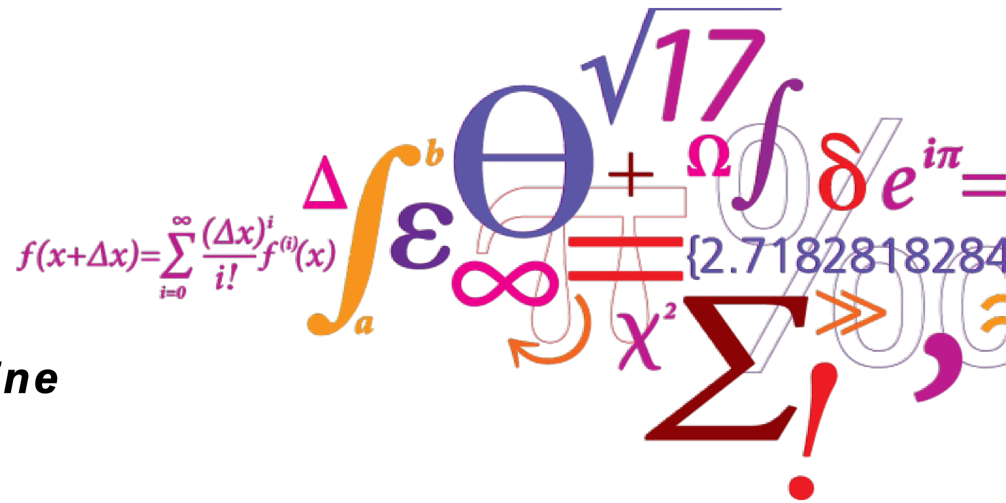
Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Larsen, T. J. (Author), Rasmussen, F. (Author), & Hansen, A. M. (Author). (2012). Gearbox modeling and simulation: Can transient loads occur internally in a planetary gearbox? How to simulate external modules fully coupled with HAWC2. Sound/Visual production (digital)

Gearbox modeling and simulation

- Can transient loads occur internally in a planetary gearbox?
- How to simulate external modules fully coupled with HAWC2

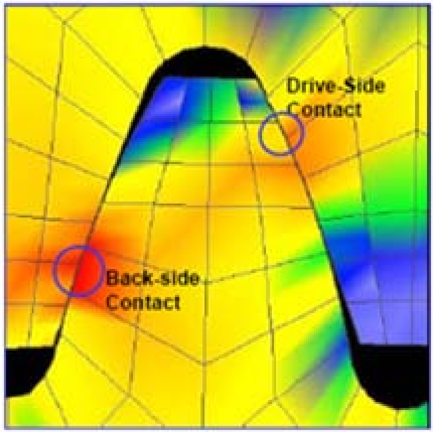
Torben Juul Larsen,
Flemming Rasmussen,
Anders M. Hansen
DTU Wind Energy, Risø Campus
tjul@dtu.dk



7th WES workshop on Wind Turbine Drive Train research
31. May 2012

Introduction

- **The simple answer is “YES”**
- **- if double contact can occur**



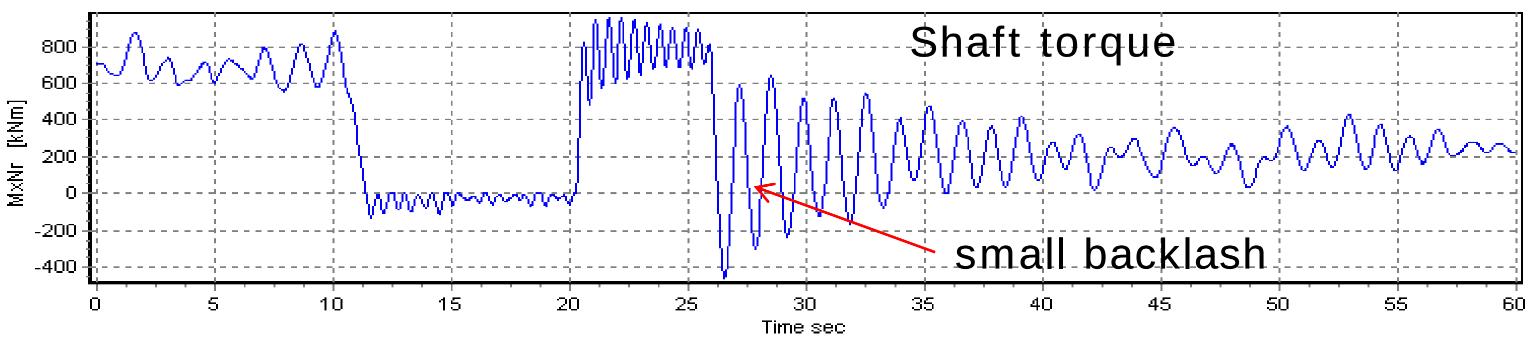
Double contact where the tooth backside is also in contact.

- Gear box failures discovered > 10 years ago on 600 – 800 kW turbines
- Modeling gearbox dynamics as an integrated part of turbine aeroelastic response and look for explanations
- Concept with gearbox acting as second main bearing

Mysteries or rumours?

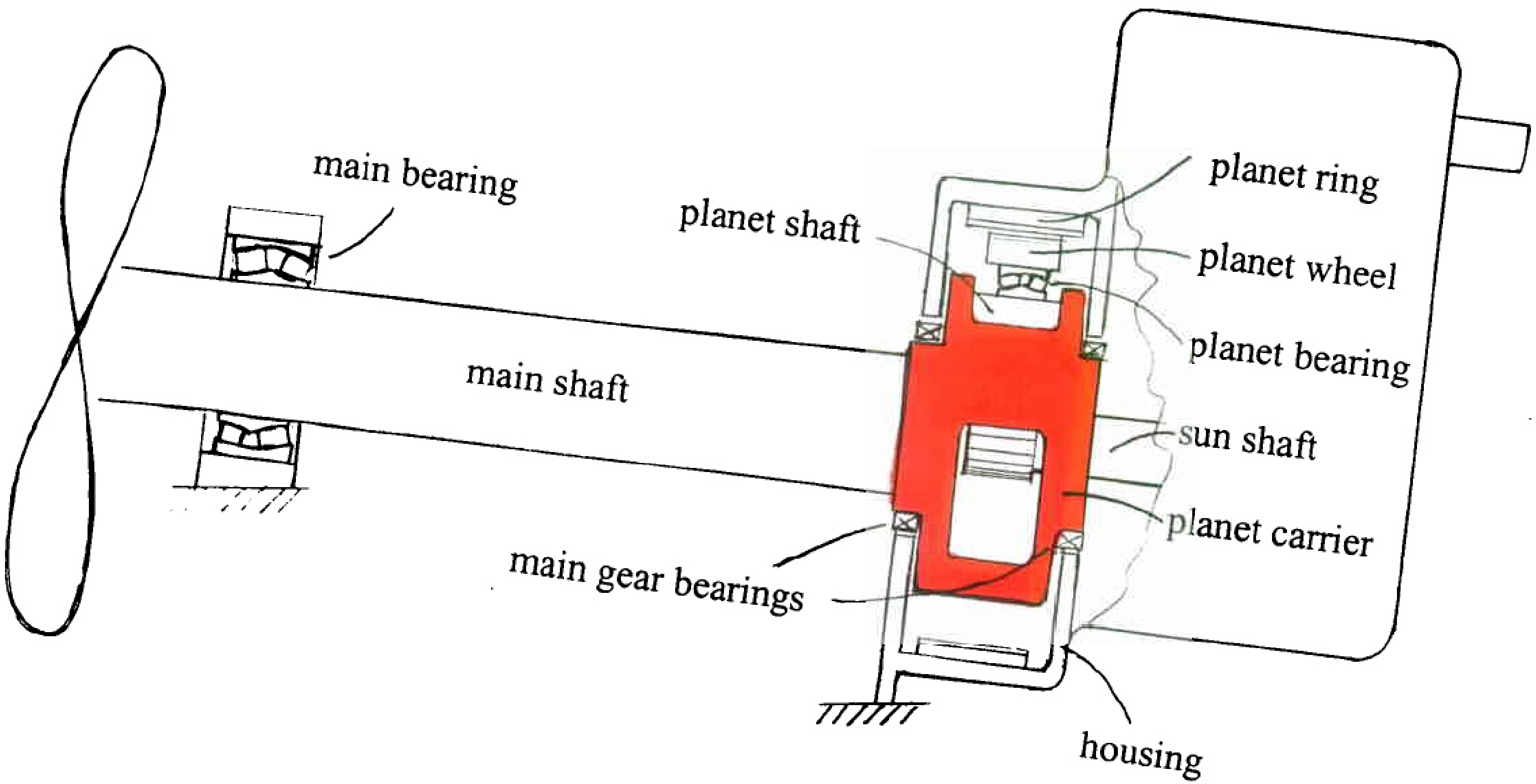
- Bearings on planetary wheels were the first to fail
- Replacing and doubling bearings made them last, however, caused the planetary ring to be warn out
- Planetary carrier bearings in some cases had high clearance
- Also high speed shaft bearings failed
- Big variety in gearbox lifetime for turbines in the same wind farm
- You could replace a broken gearbox for a turbine in a wind farm with exactly the same type, and then in some cases it would last

Braking sequence to reflect backlash:

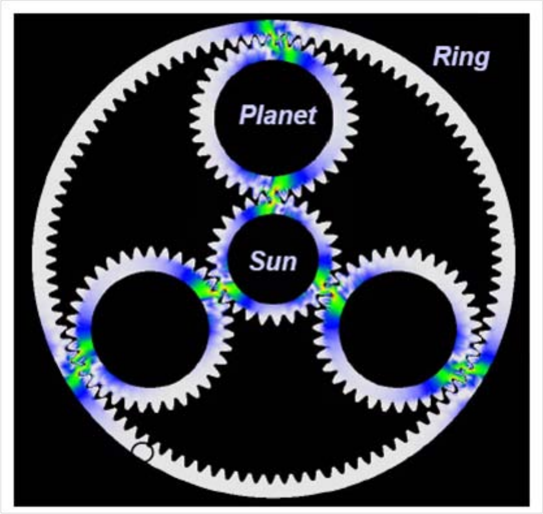


Planetary drive train

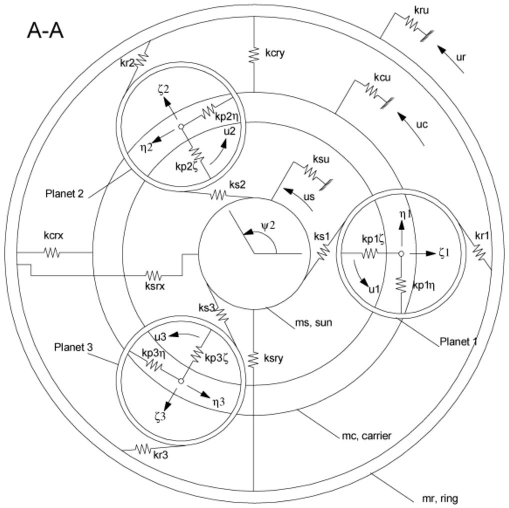
Gear box as second
main shaft bearing



Normal load pattern



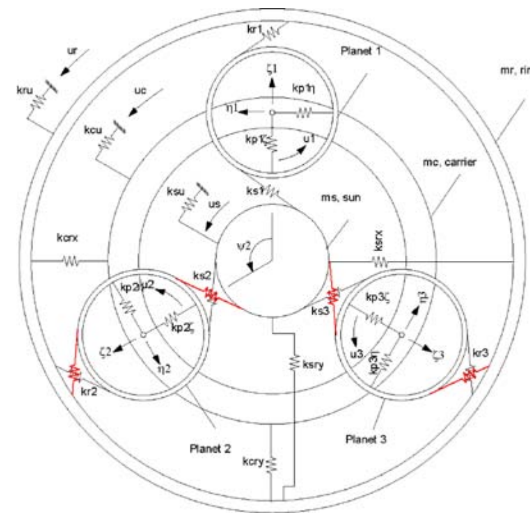
Load pattern during normal operation when loading is as expected with pressure on one tooth side only



2-dimensional model of planetary gear stage with lumped springs representing tooth and bearing stiffness.

Double contact

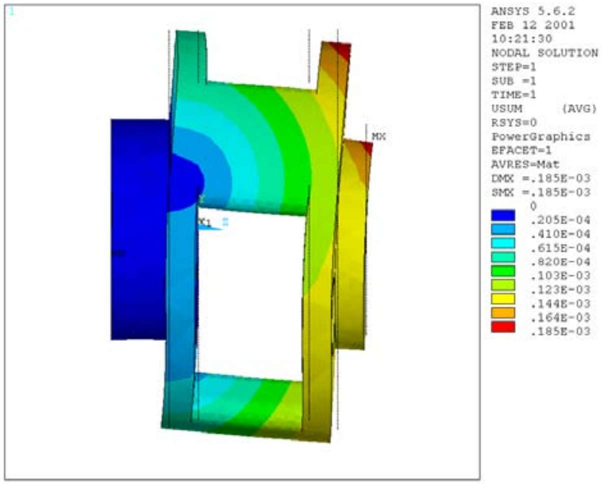
Clearance ~ 2mm



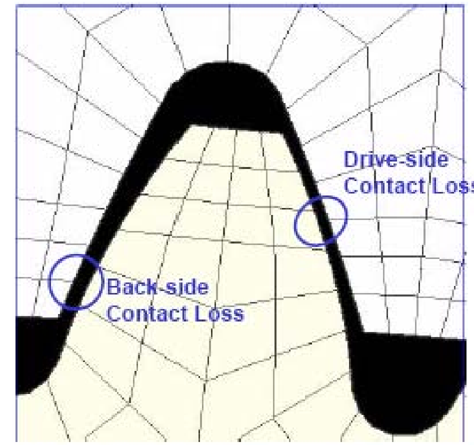
In case of a double contact the sun can be “caught” by two planets causing an uneven load distribution

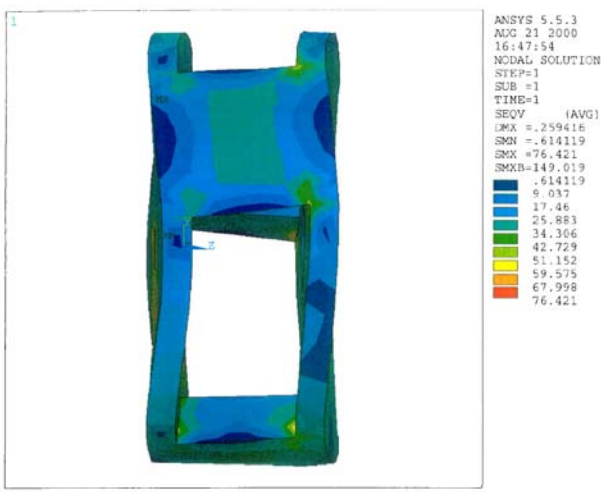
Tooth contact variations

Deformations and bearing clearance causes tooth contact variations

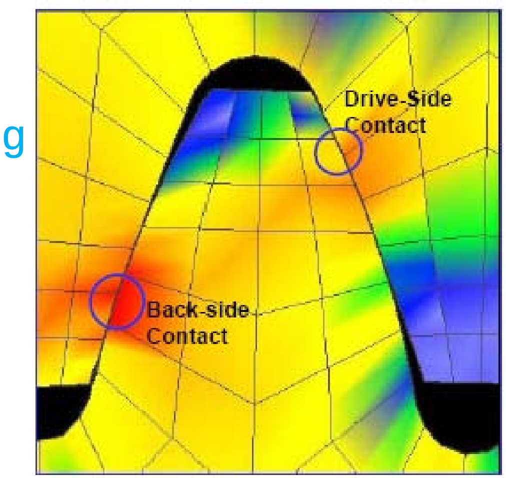


Tooth contact loss

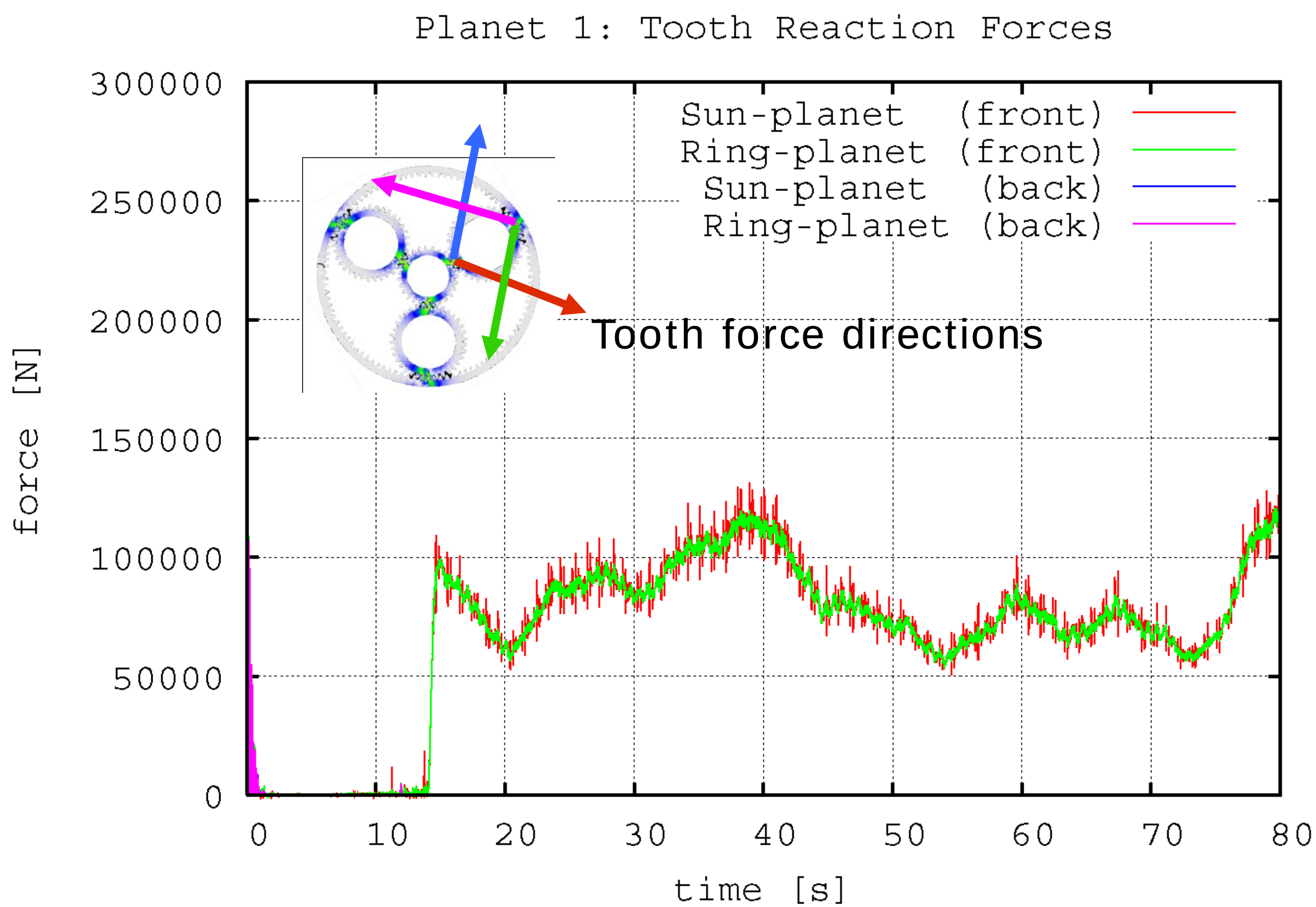




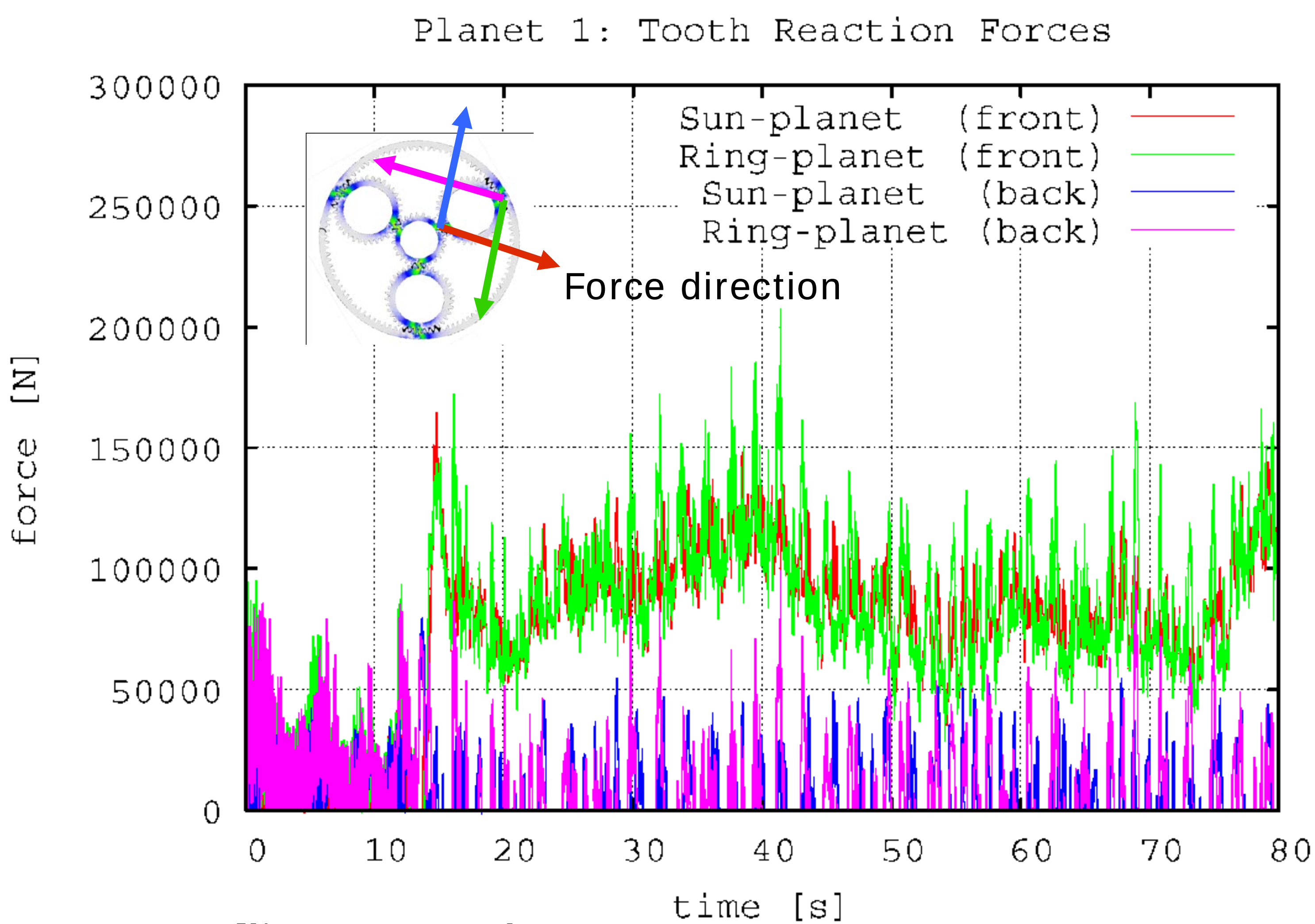
Tooth double contact or wedging



Tooth reaction forces in normal operation

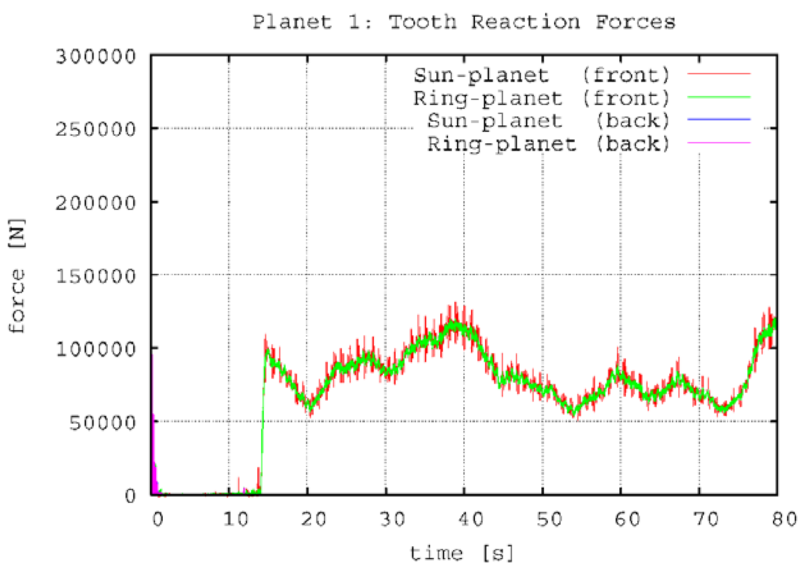


Tooth reaction forces during double contact



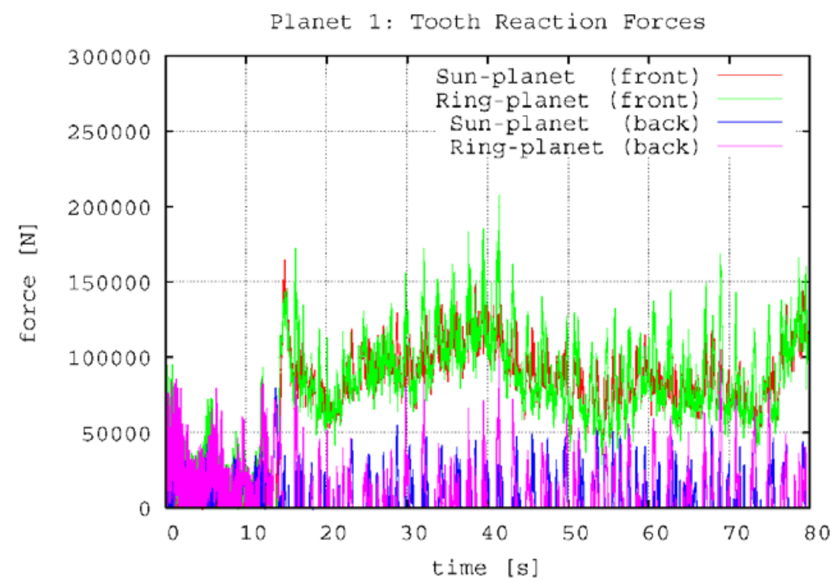
Tooth forces (planet bearing force)

Normal operation



Normal operation case without double contact. The variation in loading is caused by turbulent wind speed variation

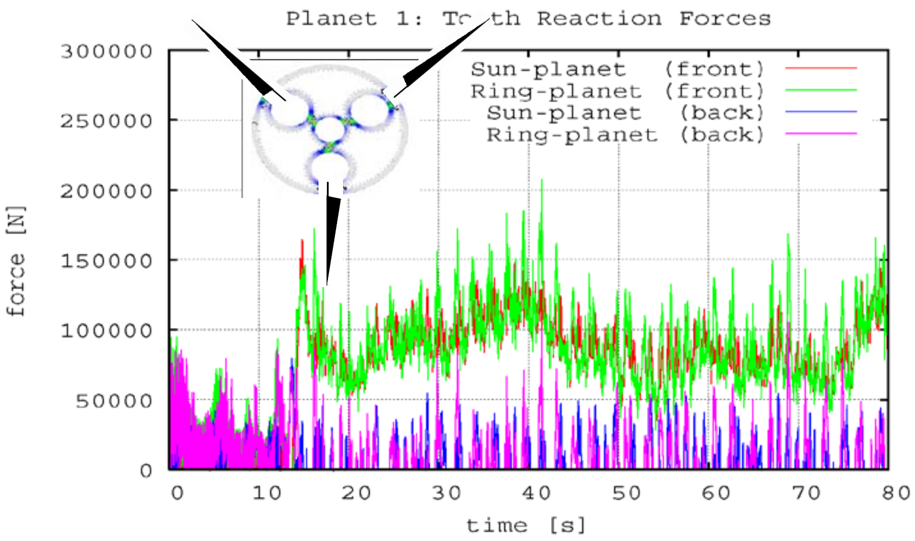
Double contact



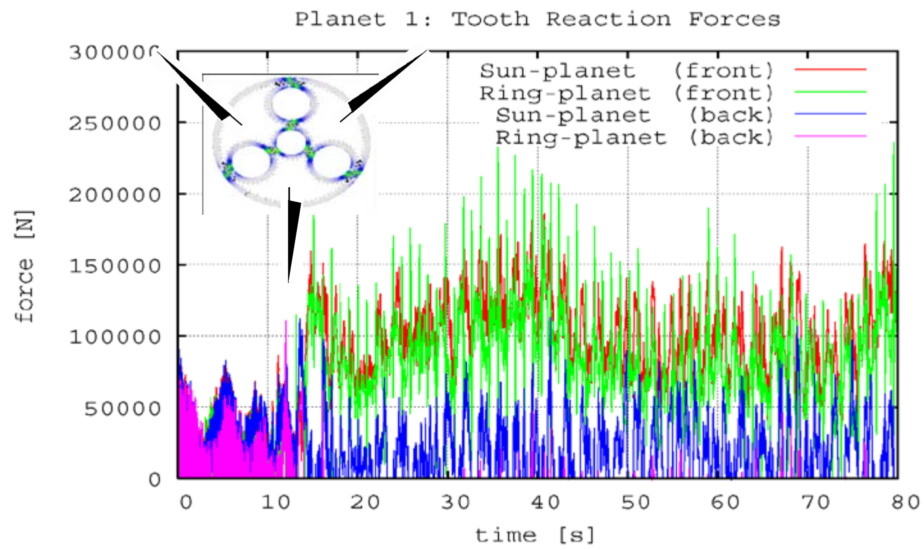
Large carrier bearing clearance is applied which causes double contact. Loads are increased significantly.

Tooth forces

Planets aligned with blades

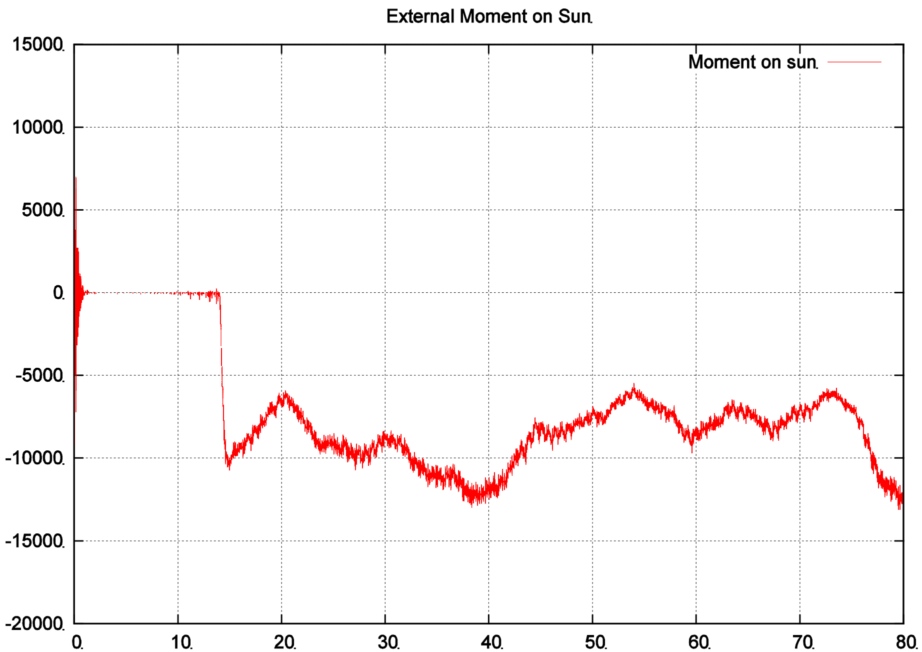


Planets rotated 60 degrees

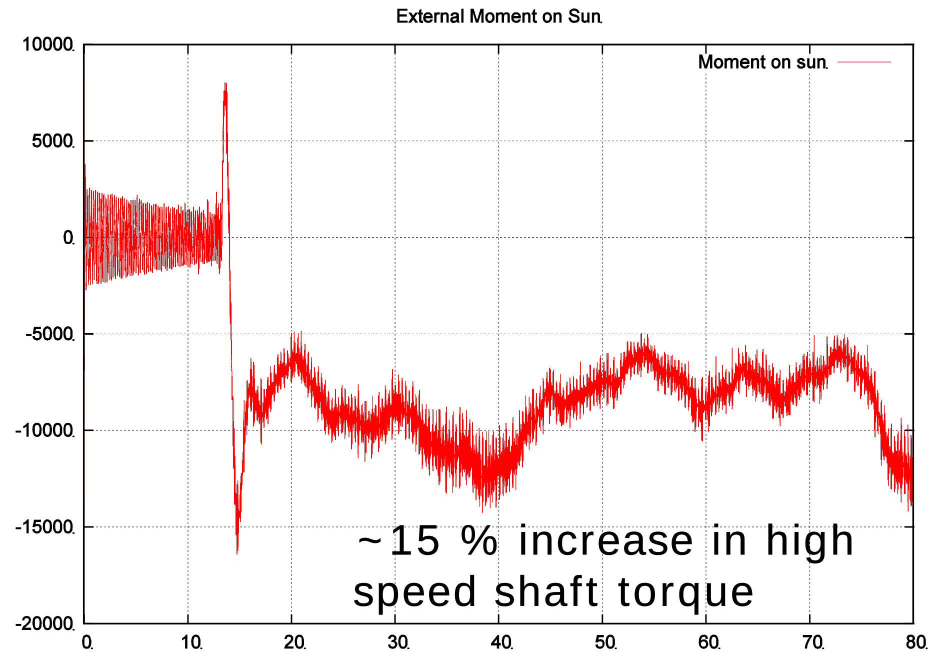


Sun torque with and without double contact

Normal operation



Double contact



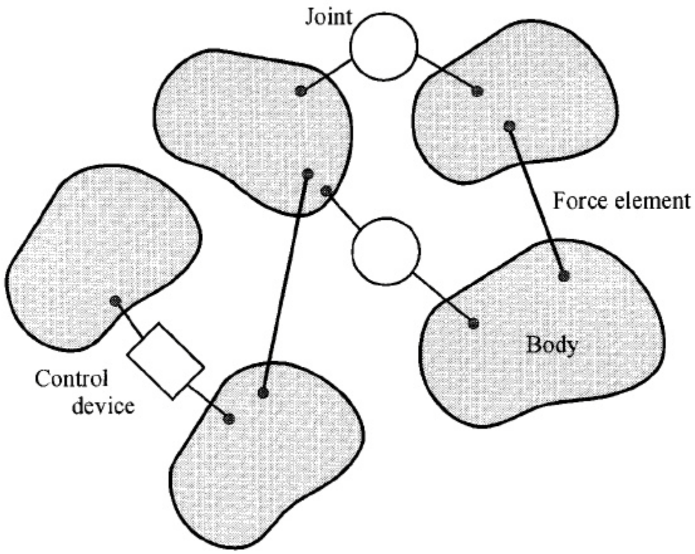
During double contact:
Bending moment loads on the main shaft causes spikes on the sun torque and thus on the high speed shaft. ~15 % increase.

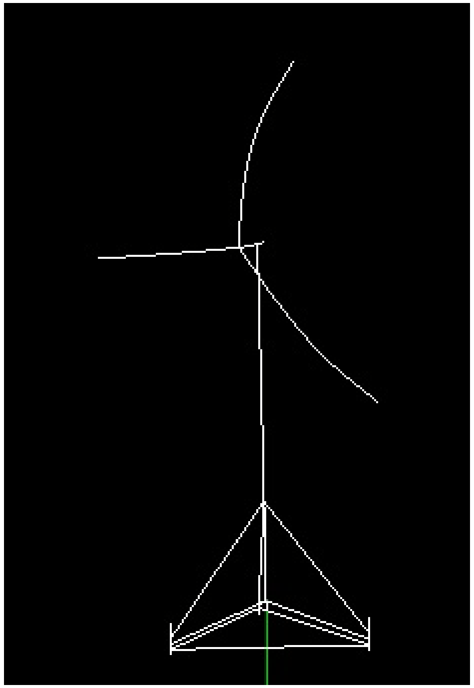
Summary on the double contact issue

- Transient loads can occur internally in a planetary gear box
  - if double contact occurs
- Tooth and bearing loads can increase with ~100 %
- The additional planet bearing loads depend upon the relative rotational position of the planets and the blades. The difference in load could be 20 %
- The load variations are transferred to the sun and the high speed shaft
- High speed shaft spikes can increase by ~15 %
- The learning is easy: In order to avoid the problem, the planetary stage should be designed with sufficient clearance to avoid double contact due to deflection from any load condition
- The backlash associated with the occurrence of negative main shaft torque is, however, also an issue.

How did we then simulate the wind turbine with gearbox?

- Basis is the aeroelastic code HAWC2 used for load simulation of wind turbines in time domain
- Structural core based on a multibody formulation
- Joints modeled by geometric constraints
- A new interface to fully couple external modules was developed (some years ago)





DLL Interface to external systems

Un-constrained EOMs:

$$\delta W = \sum_{i=1}^{N} \delta W_i = \sum_{i=1}^{N} \delta \vec{q}_i \cdot \vec{B}_i = 0$$

Constrained EOMs:

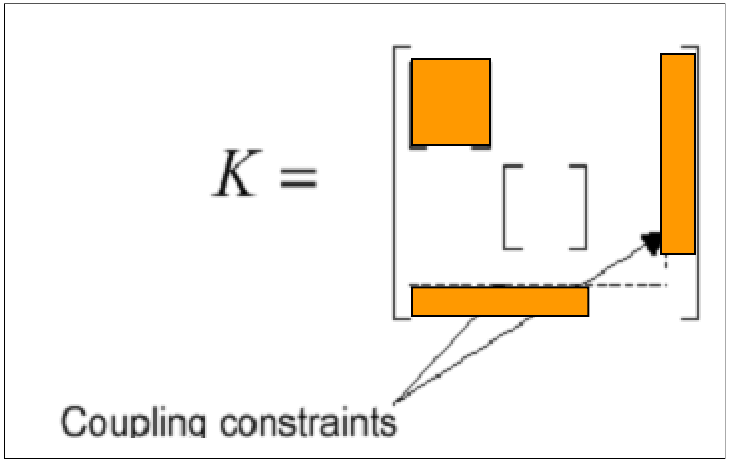
$$\delta W + \delta(\vec{\lambda} \cdot \vec{g}) = \delta W + \delta\vec{\lambda} \cdot \vec{g} + \delta\vec{g} \cdot \vec{\lambda} = 0 \quad for \quad \vec{g} = \vec{0}$$

Matrix EOMs:

$$\delta \vec{q}_i \cdot \left(\vec{B}_i + \nabla^T_{q_i} \vec{g} \cdot \vec{\lambda} \right) = 0 \quad for \ i = 1..N$$

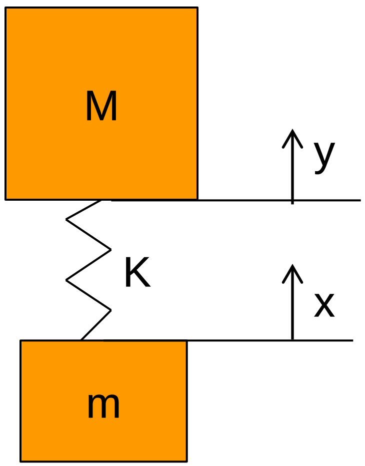
$$\delta \vec{\lambda} \cdot (\vec{g}) = 0$$

Implemented in DLL external systems !



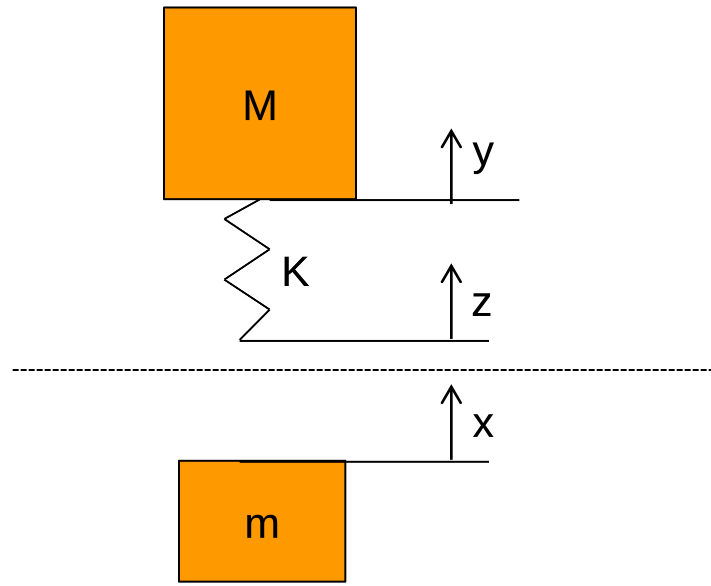
Simple example

General system



$$\begin{bmatrix} M & 0 \\ 0 & m \end{bmatrix} \begin{Bmatrix} \ddot{y} \\ \ddot{x} \end{Bmatrix} + \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{Bmatrix} y \\ x \end{Bmatrix} - \begin{Bmatrix} f_y \\ f_x \end{Bmatrix} = 0$$

HAWC2 system using constraints



$$g = z - x$$

$$\left[\begin{array}{cc|c|c} M & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & m & 0 \\ \hline 0 & 0 & 0 & 0 \end{array}\right] \begin{Bmatrix} \ddot{y} \\ \ddot{z} \\ \ddot{x} \\ \ddot{\lambda} \end{Bmatrix} + \left[\begin{array}{cc|c|c} k & -k & 0 & 0 \\ -k & k & 0 & 1 \\ \hline 0 & 0 & 0 & -1 \\ \hline 0 & 1 & -1 & 0 \end{array}\right] \begin{Bmatrix} y \\ z \\ x \\ \lambda \end{Bmatrix} - \begin{Bmatrix} f_y \\ 0 \\ f_x \\ 0 \end{Bmatrix} = 0$$

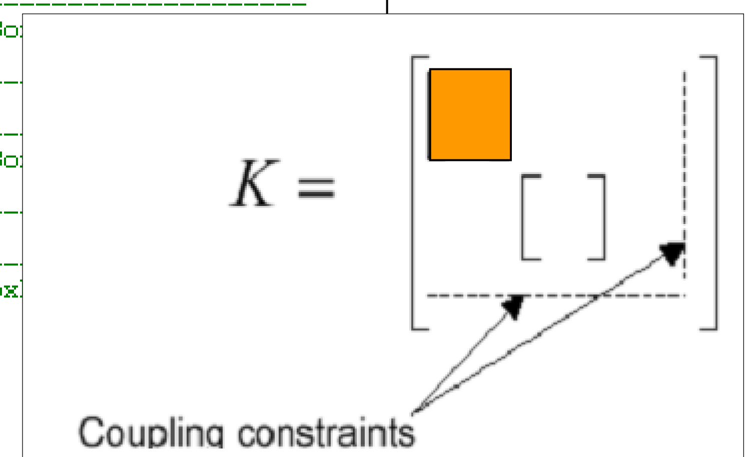
$$\delta \vec{q}_i \cdot \left(\vec{B}_i + \nabla^T_{q_i} \vec{g} \cdot \vec{\lambda} \right) = 0 \quad for\ i = 1..N$$

$$\delta \vec{\lambda} \cdot (\vec{g}) = 0$$

DLL System (programmed by the user)

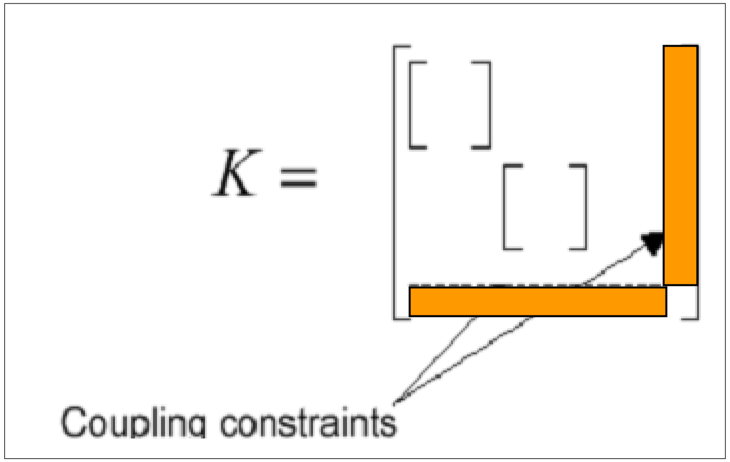
```
MODULE GearBoxDLL
!
CONTAINS
!
!+ -----------------------------------------------------------------------
subroutine GearBoxDLL_init(pwrk,nnr, nnq, nout, nvis, nheader, sdata)
!+ -----------------------------------------------------------------------
!DEC$ ATTRIBUTES ALIAS:"gearboxdll_init", DLLEXPORT::GearBoxDLL_init
!
!+ -----------------------------------------------------------------------
subroutine GearBoxDLL_initcond(pwrk,x, xdot, xdot2)
!+ -----------------------------------------------------------------------
!DEC$ ATTRIBUTES ALIAS:"gearboxdll_initcond", DLLEXPORT::GearBoxDLL_initcond
!
!+ -----------------------------------------------------------------------
subroutine GearBoxDLL_update(pwrk, time, x, xdot, xdot2, MEFF, CEFF, KEFF)
!+ -----------------------------------------------------------------------
!DEC$ ATTRIBUTES ALIAS:"gearboxdll_update", DLLEXPORT::GearBoxDLL_update
!
!+ -----------------------------------------------------------------------
SUBROUTINE GearBoxDLL_visual(pwrk, flag, iodata)
!+ -----------------------------------------------------------------------
!DEC$ ATTRIBUTES ALIAS:"gearboxdll_visual", DLLEXPORT::GearBo
!
!+ ------------------------------------------------------
SUBROUTINE GearBoxDLL_output(pwrk, time, output)
!+ ------------------------------------------------------
!DEC$ ATTRIBUTES ALIAS:"gearboxdll_output", DLLEXPORT::GearBo
!
!+ ------------------------------------------------------
SUBROUTINE GearBoxDLL_input(pwrk, time, nin, input)
!+ ------------------------------------------------------
!DEC$ ATTRIBUTES ALIAS:"gearboxdll_input", DLLEXPORT::GearBox
!
END MODULE GearBoxDLL
```

Mandatory



DLL Constraints (programmed by the user)

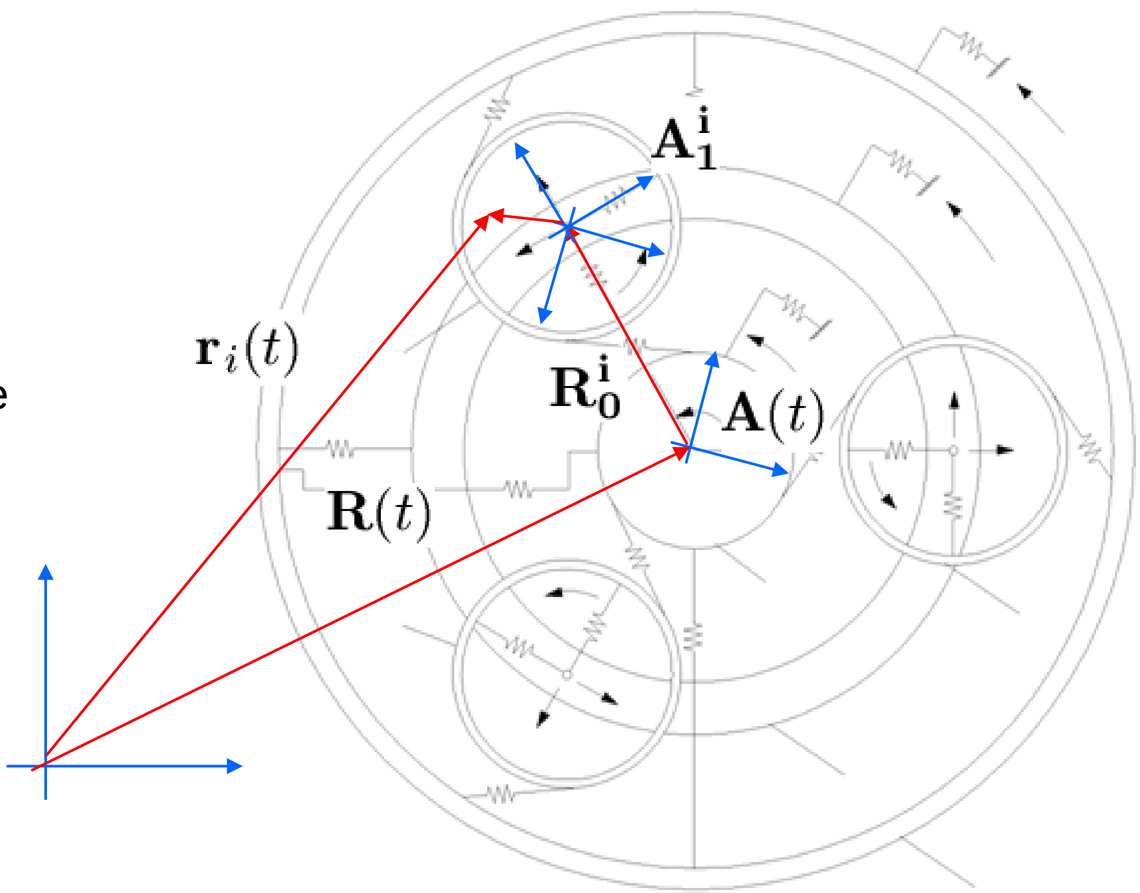
```
MODULE constraint96
CONTAINS
!
!***********************************************************************
SUBROUTINE constraint96_init(pwrk, itask, var1, var2, var3, var4, var5, strID)
!***********************************************************************
! Tasks: Initialisation, pointer setting and reading of input
!
!***********************************************************************
SUBROUTINE constraint96_update(pwrk, time)
!***********************************************************************
! Tasks: Calc. constraint vector and gradients
END MODULE
!
```



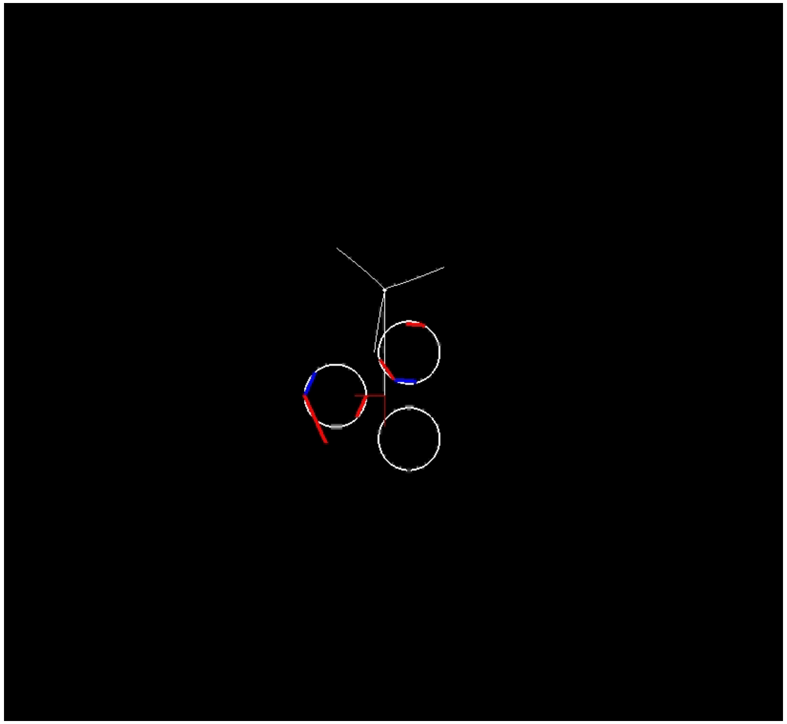
Gear Model

- Gearbox is modelled in the HAWC2 External System DLL format and coupled to the HAWC2 model via External DLL constraints.
- Each wheel is related to a floating frame of reference (6 DOFs) and has 3 individual DOFs (2 disp.+1 rot.).
  - Planets share one frame (tot. 15 DOFs)
  - Ring and sun share one frame (12 DOFs)
- Tooth and Bearing reactions are modelled as piecewise linear springs (can include one-sided stiffness and dead band)(15 DOfs).
- 42 DOFs in total

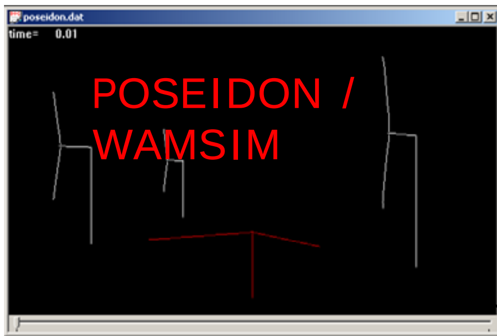
$$\mathbf{r}_i(t) = \mathbf{R}(t) + \mathbf{A}(t) \cdot \left(\mathbf{R}_0^i + \mathbf{A}_1^i \cdot \mathbf{u}_i(t) + \mathbf{A}_2^i(t) \cdot \mathbf{x}_i\right)$$

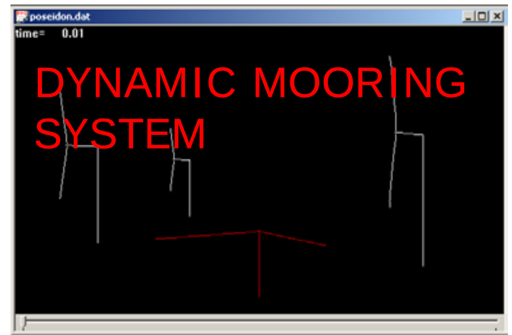


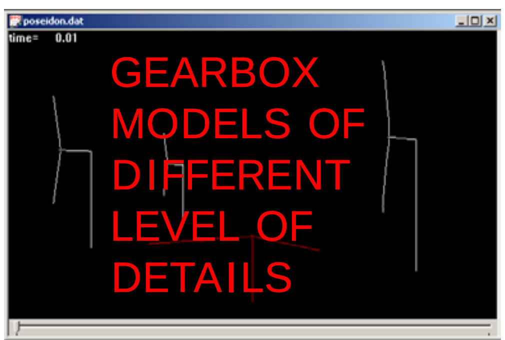
Animation of Tooth Contact forces

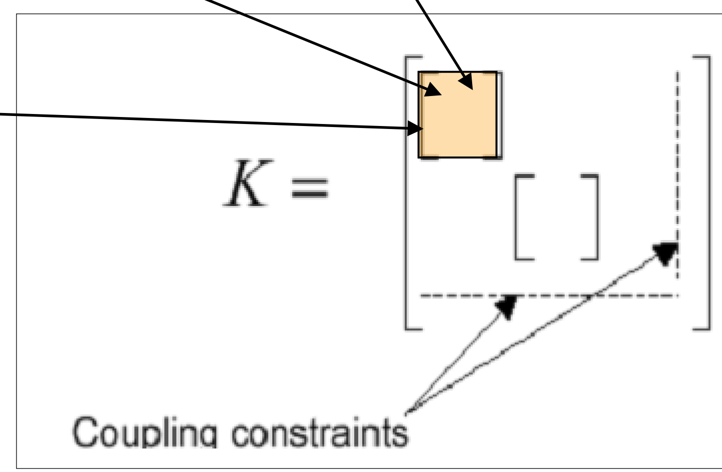


Other Examples?







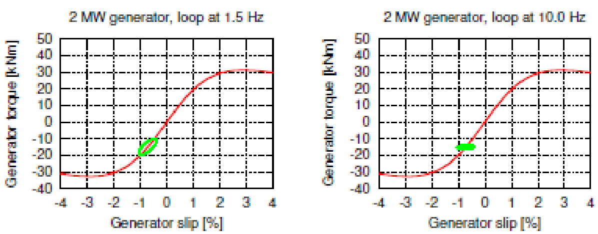


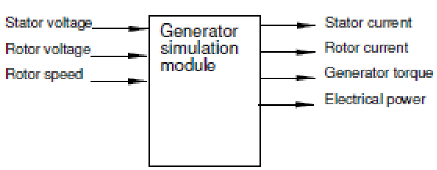
A very relevant next case
- who will give it a shot?

Risø-R-1395(EN)

Generator Dynamics in Aeroelastic Analysis and Simulations

Torben J. Larsen, Morten Hartvig Hansen,
Florin Iov





Risø National Laboratory, Roskilde, Denmark
July 2003

Flux dynamics of a doubly fed induction generator

$$\begin{Bmatrix} \dot{\psi}_s \\ \dot{\psi}_r \end{Bmatrix} = \begin{bmatrix} -\frac{R_s L_{rr}}{D} - j\omega_e & \frac{R_s L_m}{D} \\ \frac{R_r L_m}{D} & -\frac{R_r L_{ss}}{D} - j\left(\omega_e - \omega_r\right) \end{bmatrix} \begin{Bmatrix} \psi_s \\ \psi_r \end{Bmatrix} + \begin{Bmatrix} u_s \\ u_r \end{Bmatrix}$$

Electromagnetic torque

$$T = \frac{3pL_m}{2D}\mathrm{Im}\{\psi_s\bar{\psi}_r\}$$

Coupling to generator rotor inertia

$$J\,\dot{\Omega}_r = T + T_m$$